

Learning Directed-Acyclic-Graphs from Large-Scale Genomics Data

Fabio Nikolay[†], Marius Pesavento[†], George Kritikos[‡], Nassos Typas[‡]

[†]Communication Systems Group, Technische Universität Darmstadt, Germany

[‡]European Molecular Biology Laboratory (EMBL), Heidelberg, Germany

Abstract—In this paper we consider the problem of learning the genetic-interaction-map, i.e., the topology of a directed acyclic graph (DAG) of genetic interactions from noisy double knockout (DK) data. Based on a set of well established biological interaction models we detect and classify the interactions between genes. We propose a novel linear integer optimization program called the Genetic-Interactions-Detector (GENIE) to identify the complex biological dependencies among genes and to compute the DAG topology that matches the DK measurements best. Furthermore, we extend the GENIE-program by incorporating genetic-interactions-profile (GI-profile) data to further enhance the detection performance. In addition, we propose a sequential scalability technique for large sets of genes under study, in order to provide statistically stressable results for real measurement data. Finally, we show via numeric simulations that the GENIE-program as well as the GI-profile data extended GENIE (GI-GENIE)-program clearly outperform the conventional techniques and present real data results for our proposed sequential scalability technique.

Index Terms—Genetic interactions analysis, large scale gene networks, discrete optimization, big data, graph learning, multi-ple hypothesis test

I. INTRODUCTION

Genetic interaction analysis aims at uncovering the interactions among a set of genes with respect to a specified cell function of a biological system, e.g., the fitness of a specific bacteria colony. The interactions among the genes under study can be characterized by a directed-acyclic-graph (DAG)[30] where the hierarchical relationship among two genes of a DAG describes their hierarchical interaction type[7]. However DAGs cannot be observed directly but only the specified cell function under study which yields observable phenotypes. The term phenotype generally describes the specific manifestation of a biological attribute of an organism which can be observed, e.g., for bacteria a common biological attribute is the growth measured in colony size, where a specific size of the bacteria colony is a phenotype of this biological attribute. The role of the studied genes in the cell machinery, the hierarchical interaction types of the genes, as well as the DAG, which describes the latter ones, can only be learned by means of knock-out experiments where a gene or a set of genes is functionally switched off and the phenotype is observed. Traditionally, only single-knock-out (SK) experiments have been conducted but those mainly provide evidence on the importance of a single gene for the investigated cell process and do not convey much information about the interaction among the genes under study.

Recently, with the technological advances in micro arrays and the development of the synthetic-genetic-array technologies [14] new approaches have been taken that are based on large scale knock-out experiments of pairs of genes. Such double knock-out (DK) experiments are much more powerful for exploring genetic interactions since a DK phenotype of an arbitrary pair of genes generally differs considerably from the superposition of the corresponding SK phenotypes of this pair of genes. According to [7], the gene pairs can be classified into one out of five hierarchical relationship classes based on their SK and DK phenotypes. Further, based on the hierarchical relationship classes the DAG underlying the observed SK and DK phenotypes can be inferred which directly reflects the genetic interactions among the genes. In order to detect the DAG underlying the SK and DK phenotypes a variety of statistical methods based on scoring the measurements or on thresholding the genetic-interaction (GI)-profile data, which is commonly based on Pearson correlation of the SK and DK phenotypes, e.g. [1]-[6] respectively, have been developed. However, methods as presented in [1]-[6] have three considerable disadvantages: *D1*) they show poor performance in detecting the DAG underlying the observed SK and DK phenotypes *D2*) they have no ability to combine different types of side information, e.g., GI-profile data with SK and DK phenotypes, to enhance the detection quality *D3*) they cannot make use of prior knowledge in order to enhance the DAG detection quality. Especially the ability to overcome the disadvantage in *D2*) will become more important in the future, since there is a constantly increasing amount of different data types, e.g., SK and DK phenotypes, Pearson correlation based GI-profile data, other types of GI-profile data, freely available. Furthermore, the ability to overcome the deficit in *D3*), i.e., to incorporate a priori knowledge about existing results in genomics research into the DAG detection procedure, is also of great significance, since existing functional relationships among genes are increasingly better understood based on a variety of studies that constantly extend the knowledge on the cell machinery and molecular biology. Although exhibiting the above mentioned disadvantages *D1*) to *D3*), methods as those presented in [1]-[6] are the most commonly used methods to detect the DAG underlying the measured SK and DK data. Therefore, we propose the Genetic-Interactions-Detector (GENIE)-program, that is an approach based on the biological system model of [7] with which it is possible to overcome the above mentioned shortcomings of the most popular methods

as those reported in [1]-[6]. Since the hierarchical relationship classes are mutually dependent, classifying each pair of genes to a specific hierarchical relationship class corresponds to a multi-hypotheses test. Thus, we formulate this multi-hypotheses test as a linear integer optimization program, [8]-[13] in order to find the set of hierarchical relationship classes, best matching the observed SK and DK phenotypes. Based on the detected set of hierarchical relationship classes, the set of edges of the DAG which reflects the interactions among the genes can be computed. Furthermore, we propose the GI-GENIE-program where we advance the proposed GENIE-program by incorporating GI-profile data, e.g. GI-profile data based on Pearson correlation of the observed SK and DK phenotypes, into the DAG detection procedure. Due to incomplete knowledge about the true dependencies among the very most sets of genes, i.e., the true DAG of a set of genes with respect to a specific cell function is unknown or only partially known for almost all sets of genes irrespectively of the cell function under study, there is a strong interest in the genomics research community in statistically reliable statements about the topology of the DAGs underlying large sets of genes, i.e., for the empirical probability of a pair of genes to interact with each other. Towards this aim we propose a sequential technique based on the GENIE/GI-GENIE algorithms that yields statistically stressable statements about the interactions among genes from a large set of genes under study.

This paper is organized as follows. We first summarize the biological system model of [7] in Section 2, then we present in Section 3 the GENIE-program for detecting the set of hierarchical relationship classes, that represents a valid DAG and matches the DK measurements best. In Section 4 we extend the GENIE-program with GI-profile data (GI-GENIE). In Section 5, we present our scalability approach in order to obtain statistically stressable results for large sets of genes. Following Section 5, we present results for simulated data which demonstrate the performance of the GENIE and the GI-GENIE methods in Section 6. Furthermore, in Section 6 we display real data results for the scalability approach described in Section 5. Finally, we summarize in Section 7 the key parts of this paper and give a brief outlook on future work.

II. SYSTEM MODEL

In this section we provide a mathematical description of a DAG as well as its biological implications. Furthermore, we introduce the common biological terms and provide a compact description of the genetic interaction model of [7] including simple explanations on how to read and interpret a DAG of a genetic interaction map.

According to [15] a graph $\mathcal{A} = (\mathcal{V}(\mathcal{A}), \mathcal{E}(\mathcal{A}))$ is well defined by a set of nodes $\mathcal{V}(\mathcal{A}) = \{a_1, a_2, \dots, a_A\}$ and a set of edges $\mathcal{E}(\mathcal{A}) = \{\{a_1, a_A\}, \{a_2, a_A\}, \dots, \{a_A, a_1\}\}$ where $\{a_i, a_j\}$ for $a_i, a_j \in \mathcal{V}(\mathcal{A})$ denotes a directed edge from a_i to a_j and cardinality $|\mathcal{V}(\mathcal{A})| = A$ denotes the number of elements of set $\mathcal{V}(\mathcal{A})$. The operators $\mathcal{V}(\cdot)$ and $\mathcal{E}(\cdot)$ applied to graph \mathcal{A} yield the set of nodes $\mathcal{V}(\mathcal{A})$ and the set of edges $\mathcal{E}(\mathcal{A})$ respectively. We mostly address sets $\mathcal{V}(\mathcal{A})$ and $\mathcal{E}(\mathcal{A})$ by \mathcal{G}_A

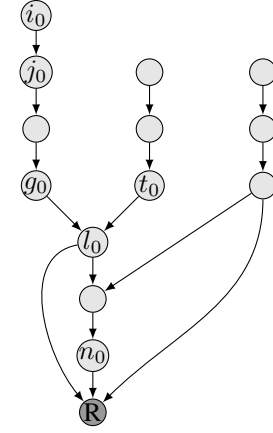


Fig. 1: DAG \mathcal{D}_0 of 13 genes and root node R

and \mathcal{E}_A respectively for the sake of notational convenience, i.e., $\mathcal{A} = (\mathcal{G}_A, \mathcal{E}_A)$. Assume that there is a path P from node $a_i \in \mathcal{G}_A$ to node $a_j \in \mathcal{G}_A$ in graph \mathcal{A} , i.e., a directed connection from node $a_i \in \mathcal{G}_A$ to node $a_j \in \mathcal{G}_A$. Then path P is described by the concatenation of nodes being passed through on the way from node $a_i \in \mathcal{G}_A$ to node $a_j \in \mathcal{G}_A$, i.e., $P = a_i \dots a_j$ and $\mathcal{V}(P) = \{a_i, \dots, a_j\}$ denotes the set of nodes of path P [15].

The functional dependencies among a set of genes $\mathcal{G} = \{g_1, \dots, g_G\}$, with $G = |\mathcal{G}|$ elements, for a given cell process and specie can be characterized by a genetic-interaction-map (GI-map, [17]-[20]) which is essentially a DAG with a common root node, i.e., the reporter level R, [31]. In particular, an arbitrary DAG \mathcal{D} can be described as a graph $\mathcal{D} = (\mathcal{G}_D, \mathcal{E}_D)$ with the set of nodes $\mathcal{G}_D = \{\mathcal{G} \cup R\}$ and the set of directed edges $\mathcal{E}_D = \{\{g_i, g_j\}, \dots, \{g_j, g_l\}\}$. As the genetic interactions can only be observed through the reporter, all edges are always orientated in such a way that each path parting from any arbitrary gene $g_i \in \mathcal{G}$ always terminates in the root node R and any gene appears on the path at most once, i.e., there exist no cycles in the graph. Hence, the DAG \mathcal{D} is always connected via its root node R. For the sake of notational convenience, in most cases we write gene i when addressing gene g_i , [31]. The reporter node R is an artificial node, i.e., not a gene, in the concept of a DAG and represents the measured phenotype of the specific cell process under study.

To provide a better understanding of the information encoded in a DAG we state a simple example, which is similar to the one in [31], based on DAG \mathcal{D}_0 displayed in Fig. 1. In \mathcal{D}_0 there exists a direct edge from gene i_0 to gene j_0 , i.e. $\{i_0, j_0\} \in \mathcal{E}_{\mathcal{D}_0}$, which indicates that the activity of gene i_0 controls the activity of gene j_0 . Hence, gene i_0 only affects the phenotype via gene j_0 and not directly. We emphasize that in this model the existence of edge $\{i_0, j_0\}$ in the DAG only describes the hierarchical functional dependency between genes i_0 and j_0 and not the quantitative effect of gene i_0 on gene j_0 .

Denote $R(i) \in \mathbb{R}$ as the phenotype for a single gene $i \in \mathcal{G}$

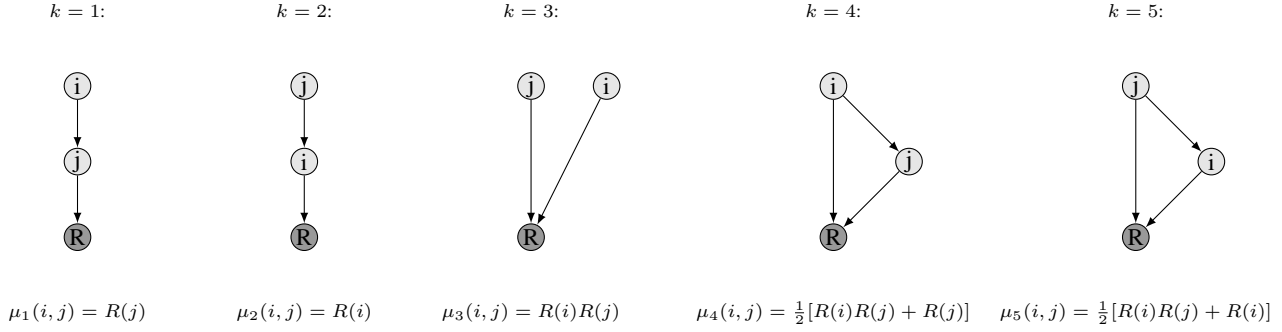


Fig. 2: Possible hierarchical relationship classes between two arbitrary genes i, j of DAG \mathcal{D} according to [7]

functionally switched off. In the same fashion we define the phenotype for the DK of genes $i, j \in \mathcal{G}$: $j > i$ as $R(i, j) \in \mathbb{R}$. Based on the DK phenotypes the GI-profile similarity of genes $i, j \in \mathcal{G}$: $j > i$ is computed as the Pearson correlation between all DK phenotypes which involve gene i and all DK phenotypes which involve gene j , i.e., the GI-profile similarity $\rho(i, j)$ is given by

$$\rho(i, j) = \frac{\sum_{l \in \mathcal{G} \setminus \{i, j\}} (R(i, l) - \bar{R}(i)) (R(j, l) - \bar{R}(j))}{\sqrt{\sum_{l \in \mathcal{G} \setminus \{i, j\}} (R(i, l) - \bar{R}(i))^2 \sum_{l \in \mathcal{G} \setminus \{i, j\}} (R(j, l) - \bar{R}(j))^2}} \quad (1)$$

with $\bar{R}(i) = \frac{1}{G-2} \sum_{l \in \mathcal{G} \setminus \{i, j\}} R(i, l)$, $\bar{R}(j) = \frac{1}{G-2} \sum_{l \in \mathcal{G} \setminus \{i, j\}} R(j, l)$ and $R(i, l) = R(l, i) \forall i, l \in \mathcal{G}$: $l > i$. We remark that the GI-profile similarity data $\rho(i, j)$ does not have to be computed according to Eq. (1) necessarily. It can also be extracted from a data base where a priori knowledge about the set of genes under study, i.e., \mathcal{G} , is stored. In genomics research it is a common assumption that if there is an edge between two genes $i, j \in \mathcal{G}$: $j > i$ in DAG \mathcal{D} , i.e., there is an interaction between genes $i, j \in \mathcal{G}$: $j > i$ in DAG \mathcal{D} , then the GI-profile similarity $\rho(i, j)$ is very likely to be high. Furthermore according to [7] we assume that each pair of genes $i, j \in \mathcal{G}$: $j > i$ belongs to exactly one out of five hierarchical relationship classes that are characterized in Fig. 2. The hierarchical relationship classes $k \in \mathcal{K} = \{1, \dots, 5\}$ are defined according to the model $\mu_k(i, j)$ in which the single knock-out phenotypes $R(i)$ and $R(j)$ are related with the DK phenotype $R(i, j)$. If the gene pair $i, j \in \mathcal{G}$: $j > i$ belongs to the hierarchical relationship class k then the observed DK phenotype $R(i, j)$ is described by the model $\mu_k(i, j)$ provided in Fig. 2. We remark that the five hierarchical dependency graphs in Fig. 2 do not reflect the absolute adjacency relations, but the hierarchical relations between genes i, j in DAG \mathcal{D} . Hence, given that two genes i, j of DAG \mathcal{D} are in class k , we cannot conclude that genes i, j are directly arranged in DAG \mathcal{D} as displayed by the depiction of class k in Fig. 2. This follows from the fact that the description of the hierarchical relationship classes provided in Fig. 2 only contains relative topology information about two genes i, j in

DAG \mathcal{D} .

In the following and in addition to [7], we provide, for clarity of presentation, a formal description of the hierarchical relationship classes depicted in Fig. 2 using a graph theoretical representation. Assume that there are I paths $\mathcal{P}_{i, \tau}$, for $\tau \in \{1, \dots, I\}$, from gene i to the reporter node R in DAG \mathcal{D} and the set \mathcal{P}_i containing all such paths is defined as $\mathcal{P}_i = \{\mathcal{P}_{i, 1}, \dots, \mathcal{P}_{i, I}\}$. Furthermore, set $\mathcal{P}_j = \{\mathcal{P}_{j, 1}, \dots, \mathcal{P}_{j, J}\}$ contains all J paths from gene j to the reporter node R in DAG \mathcal{D} . Given gene pair $i, j \in \mathcal{G}$: $j > i$ in DAG \mathcal{D} , then pair i, j belongs to the hierarchical relationship class $k \in \mathcal{K}$ if and only if condition C_k as defined below is satisfied:

$$C_1 : \quad \forall \mathcal{P}_{i, \tau} \in \mathcal{P}_i : j \in V(\mathcal{P}_{i, \tau}) \quad (2a)$$

$$C_2 : \quad \forall \mathcal{P}_{j, \tau} \in \mathcal{P}_j : i \in V(\mathcal{P}_{j, \tau}) \quad (2b)$$

$$C_3 : \quad \left(\forall \mathcal{P}_{i, \tau} \in \mathcal{P}_i : j \notin V(\mathcal{P}_{i, \tau}) \right) \wedge \left(\forall \mathcal{P}_{j, \tau} \in \mathcal{P}_j : i \notin V(\mathcal{P}_{j, \tau}) \right) \quad (2c)$$

$$C_4 : \quad \left(\exists \mathcal{P}_{i, \tau} \in \mathcal{P}_i : j \notin V(\mathcal{P}_{i, \tau}) \right) \wedge \left(\exists \mathcal{P}_{i, \tau} \in \mathcal{P}_i, \mathcal{P}_{j, \tau} \in \mathcal{P}_j : V(\mathcal{P}_{j, \tau}) \subset V(\mathcal{P}_{i, \tau}) \right) \quad (2d)$$

$$C_5 : \quad \left(\exists \mathcal{P}_{j, \tau} \in \mathcal{P}_j : i \notin V(\mathcal{P}_{j, \tau}) \right) \wedge \left(\exists \mathcal{P}_{i, \tau} \in \mathcal{P}_i, \mathcal{P}_{j, \tau} \in \mathcal{P}_j : V(\mathcal{P}_{i, \tau}) \subset V(\mathcal{P}_{j, \tau}) \right) \quad (2e)$$

As stated in condition C_1 in (2a), two genes $i, j \in \mathcal{G}$: $j > i$ in DAG \mathcal{D} belong to the hierarchical relationship class $k = 1$, if all paths from gene i to the reporter node R pass through gene j . Hence, gene j is always an element of the set of nodes of each path $\mathcal{P}_{i, \tau} \in \mathcal{P}_i$ from gene i to the reporter node R , i.e., $j \in V(\mathcal{P}_{i, \tau})$ for all paths $\mathcal{P}_{i, \tau}$ from gene i to the reporter node R . With the same line of argument as used in (2a), two genes $i, j \in \mathcal{G}$: $j > i$ in DAG \mathcal{D} belong to the hierarchical relationship class $k = 2$ if condition C_2 in (2b) is satisfied.

Two genes $i, j \in \mathcal{G} : j > i$ in DAG \mathcal{D} belong to the hierarchical relationship class $k = 3$ and are considered to be independent from each other if condition C_3 in (2c) is satisfied which states that there is no path $P_{i,\tau}$ from gene i to the reporter node R that passes through gene j as well as there is no path $P_{j,\bar{\tau}}$ from gene j to the reporter node R that passes through gene i . As stated in (2d), two genes $i, j \in \mathcal{G} : j > i$ in DAG \mathcal{D} belong to the hierarchical relationship class $k = 4$ if there is at least one path $P_{i,\tau}$ from gene i to the reporter node R which does not pass through gene j as well as for all paths $P_{j,\bar{\tau}} \in \mathcal{P}_j$ there is always a path $P_{i,\tau} \in \mathcal{P}_i$ that is a super-path of the respective $P_{j,\bar{\tau}} \in \mathcal{P}_j$. With the same line of argument as used in (2d), two genes $i, j \in \mathcal{G} : j > i$ in DAG \mathcal{D} belong to the hierarchical relationship class $k = 5$ if condition C_5 in (2e) is satisfied.

To illustrate this, let us consider the example DAG \mathcal{D}_0 of Fig. 1. All paths from gene i_0 to node R pass through gene j_0 , i.e., they are in a linear pathway with gene i_0 upwards of gene j_0 . Thus the pair of genes i_0, j_0 belongs to class $k = 1$. Note that with the same line of argument, we conclude that also genes i_0 and l_0 belong to relationship class $k = 1$. Since all paths from gene i_0 to the reporter level R do not pass through gene t_0 and all paths from gene t_0 to the reporter level do not pass through gene i_0 , genes i_0 and t_0 belong to the hierarchical relationship class $k = 3$ as given in Fig. 2, which states that genes i_0 and t_0 are independent of each other and the DK phenotype amounts to $R(i_0, t_0) = \mu_3(i_0, t_0)$. Finally, let us investigate the hierarchical relation between genes t_0 and n_0 in DAG \mathcal{D}_0 . Obviously, gene t_0 has (at least) one path to node R which does not pass through gene n_0 , i.e., genes only having paths to R that do not pass through gene n_0 do not affect the activity of gene n_0 . Since there is (at least) one other path from gene t_0 to R passing through gene n_0 , we can reason that genes t_0 and n_0 belong to class $k = 4$. Generally, there are strong implications among the hierarchical relationship classes of [7], i.e., if some pairs belong to a specific class then this has strong implications for all other pairs. Let us consider the case that DAG \mathcal{D}_0 was not known and only the hierarchical relationship classes for genes i_0 and j_0 , i.e., genes i_0 and j_0 belong to class $k = 1$, as well as the hierarchical relationship class for genes i_0 and g_0 , i.e., genes i_0 and g_0 belong to class $k = 1$ were available. By definition of the hierarchical dependency graphs in Fig. 2 and the assumptions, that genes i_0 and j_0 belong to class $k = 1$ as well as that genes i_0 and g_0 belong to class $k = 1$, we conclude that all paths from gene i_0 to R pass through genes j_0 and g_0 . Thus, either all paths from gene g_0 to R pass through gene j_0 , or all paths from gene j_0 to R pass through gene g_0 . Consequently, genes j_0 and g_0 either belong to the hierarchical relationship class $k = 1$, or $k = 2$.

As we have emphasized by the example above, generally if the hierarchical relationship class is known for two arbitrary genes $i, j \in \mathcal{G} : j > i$ as well as for another pair $i, l \in \mathcal{G} : l > i$, then this has strong logical implications on the hierarchical relationship classes genes $j, l \in \mathcal{G} : l > j$ can belong to. Since we can interpret the classification of the pairs of genes

$i, j \in \mathcal{G} : j > i$, based on their observed SK and DK phenotypes $R(i), R(j)$ and $R(i, j)$, respectively, to exactly one out of the five hierarchical relationship classes as a coupled multi-hypotheses test, we address this problem in Section 3 by a linear integer optimization program. The proposed linear integer optimization program identifies the most consistent set of hierarchical relationship classes, i.e., the set of hierarchical relationship classes that represents a valid DAG and matches best the DK measurements with respect to the logical coupling between the classes. Furthermore, in Section 4 we extend the GENIE-program proposed in Section 3 by incorporating GI-profile data in order to jointly detect the most consistent set of hierarchical relationship classes and the corresponding DAG topology.

III. GENIE-ALGORITHM

In this section, we formulate the problem of classifying the gene pairs $i, j \in \mathcal{G} : j > i$ into the classes of hierarchical relationships based on the observed SK and DK phenotype values as a linear integer optimization program. Furthermore, we translate the logical implications among the hierarchical relationship classes into constraints that ensure that the detected set of hierarchical relationship classes represents a valid graph. That is the detected set of hierarchical relationship classes represents a graph which is a DAG as defined in Section 2. Finally, we propose a policy to derive an estimate $\hat{\mathcal{E}}_{\mathcal{D}}$ of the true set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG \mathcal{D} based on the detected set of hierarchical relationship classes.

A. Hierarchical Relationship Class Detection

In order to quantify the mismatch between the measured DK phenotypes $R(i, j)$ and the phenotype model $\mu_k(i, j)$ of class $k \in \mathcal{K}$ according to Fig. 2, under the hypothesis that the gene pairs $i, j \in \mathcal{G} : j > i$ belong to class k given their respective SK values, we propose a simple quadratic score [7], [31], as given in Eq. (3):

$$s_k(i, j) = (R(i, j) - \mu_k(i, j))^2 \quad k \in \mathcal{K}, \quad \forall i, j \in \mathcal{G} : j > i \quad (3)$$

Let us define the following selection variables

$$\alpha_k(i, j) = \begin{cases} 1 & \text{if } i, j \text{ are in class } k \\ 0 & \text{else} \end{cases} \quad k \in \mathcal{K}, \quad \forall i, j \in \mathcal{G} : j > i \quad (4)$$

We remark that every DAG \mathcal{D} can be represented by a set of hierarchical relationship classes which directly corresponds to a set of selection variables $A^{\mathcal{D}} = \bigcup_{\forall i, j \in \mathcal{G} : j > i} \{\alpha_1^{\mathcal{D}}(i, j), \dots, \alpha_5^{\mathcal{D}}(i, j)\}$. The GENIE-algorithm of classifying the gene pairs $i, j \in \mathcal{G} : j > i$ into the set of hierarchical relationship classes, that represents a valid DAG and matches the observed SK and DK phenotypes best can be formulated as

O_{GENIE} :

$$\min_{\{\alpha_k(i,j)\}} \sum_{i=1}^G \sum_{j=i+1}^G \left(\sum_{k=1}^{|\mathcal{K}|} s_k(i,j) \alpha_k(i,j) \right) \quad (5a)$$

$$\text{s. t. } \alpha_k(i,j) \in \{0,1\} \quad \forall k \in \mathcal{K}, \quad \forall i,j \in \mathcal{G} : j > i \quad (5b)$$

$$\sum_{k=1}^{|\mathcal{K}|} \alpha_k(i,j) = 1, \quad \forall i,j \in \mathcal{G} : j > i \quad (5c)$$

$$\mathcal{L} \implies \text{additional topology constraints} \quad (5d)$$

where $A^{\text{O}_{\text{GENIE}}} = \bigcup_{\forall i,j \in \mathcal{G} : j > i} \{\alpha_1^{\text{O}_{\text{GENIE}}}(i,j), \dots, \alpha_5^{\text{O}_{\text{GENIE}}}(i,j)\}$ denotes the solution of program O_{GENIE} in (5) and the set of best matching selection variables $A^{\text{O}_{\text{GENIE}}}$ corresponds to the most consistent pattern of hierarchical relationship classes. Problem O_{GENIE} in (5) is a linear integer program which can be solved efficiently by BB-methods [21]-[27]. The objective of problem O_{GENIE} is to minimize the overall mismatch in classifying each gene pair $i,j \in \mathcal{G} : j > i$ to one out of five hierarchical relationship classes. The constraints in (5b) reflect the binary nature of the selection variables $\alpha_k(i,j)$, $\forall i,j \in \mathcal{G} : j > i$, $\forall k \in \mathcal{K}$, while (5c) represents a multiple choice constraint to enforce that the gene pairs i,j are only classified to one out of the five hierarchical relationship classes. The set \mathcal{L} in (5d) is comprised of additional constraints to ensure that the detected set of selection variables $A^{\text{O}_{\text{GENIE}}}$ always represents a valid graph, i.e., a DAG. In the following, we exemplarily derive topology constraints in set \mathcal{L} . In order to identify the numerous logical dependencies among the selection variables $\alpha_k(i,j)$, $k \in \mathcal{K}$ for all $i,j \in \mathcal{G} : j > i$ we proceed in the following way. We first fix the assumption that genes $i,j \in \mathcal{G} : j > i$ belong to class $k = 1$, i.e., $\alpha_1(i,j) = 1$. Further we assume that genes $i,l \in \mathcal{G} : l > i$ belong to class k' , i.e. $\alpha_{k'}(i,l) = 1$. Then we derive the set of classes \mathcal{K}'' that genes $j,l \in \mathcal{G} : l > j$ can belong to under the assumptions made. In the following, we have formulated the logical dependencies among the selection variables for $\alpha_1(i,j) = 1$, i.e., the case that gene i is linearly upstream of gene j , as linear integer inequalities defined in constraints (6a)-(6e) and summarize them as set \mathcal{L}_1

$$\mathcal{L}_1 = \left\{ \begin{aligned} &\alpha_1(j,l) + \alpha_2(j,l) \geq \alpha_1(i,j) + \alpha_1(i,l) - 1 \quad (6a) \\ &\alpha_2(j,l) \geq \alpha_1(i,j) + \alpha_2(i,l) - 1 \quad (6b) \\ &\alpha_2(j,l) + \alpha_3(j,l) + \alpha_5(j,l) \geq \\ &\alpha_1(i,j) + \alpha_3(i,l) - 1 \quad (6c) \\ &\alpha_2(j,l) + \alpha_4(j,l) \geq \alpha_1(i,j) + \alpha_4(i,l) - 1 \quad (6d) \\ &\alpha_5(j,l) + \alpha_2(j,l) \geq \alpha_1(i,j) + \alpha_5(i,l) - 1 \quad (6e) \end{aligned} \right\} \forall i,j,l \in \mathcal{G}_{\mathcal{D}} : l > j > i$$

where constraints (6a)-(6e) are affine after the continuous relaxation of the selection variables $\alpha_k(i,j)$, $\forall i,j \in \mathcal{G} : j > i$. To explain the origin and the functionality of the constraints in \mathcal{L}_1 , let us further define a sub-genetic-interactions-map (SMAP) \mathcal{S} , [31], as given in Fig. 3 according to the following

definition where we adopt the graph notation of [15]:

Definition:

Given a non-empty set of edges \mathcal{E}_{in} and a non-empty set of edges \mathcal{E}_{out} . Graph $\mathcal{S} = (\mathcal{G}_{\mathcal{S}}, \mathcal{E}_{\mathcal{S}})$, with set of nodes $\mathcal{G}_{\mathcal{S}}$ and set of edges $\mathcal{E}_{\mathcal{S}}$, is a SMAP if the following conditions are fulfilled: *i*) the graph \mathcal{S} is acyclic and directed *ii*) there $\exists e_{\text{in}} \in \mathcal{E}_{\text{in}}, e_{\text{out}} \in \mathcal{E}_{\text{out}}$ such that each path P through graph \mathcal{S} incides \mathcal{S} via egde e_{in} and leaves graph \mathcal{S} via edge e_{out} .

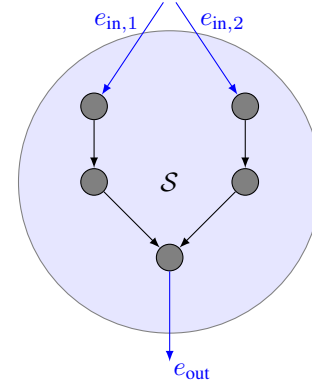


Fig. 3: Example SMAP \mathcal{S}

DAG \mathcal{D}_1 , as displayed in Fig. 4, consists of genes i,j and SMAPs \mathcal{S}_1 and \mathcal{S}_2 . Obviously, genes i,j belong to class $k = 1$, i.e., $\alpha_1(i,j) = 1$. Furthermore, all genes $l \in \mathcal{G}_{\mathcal{D}_1} \setminus \{R\} : l > j > i$ for which $\alpha_1(i,l) = 1$ must be either located in SMAP \mathcal{S}_1 or \mathcal{S}_2 . Thus, it follows from DAG \mathcal{D}_1 in Fig. 4 that the gene pair j,l is either in hierarchical relationship class $k = 1$ or $k = 2$, i.e., $\alpha_1(j,l) = 1$ or $\alpha_2(j,l) = 1$.

This logical implication is directly reflected by constraint (6a). Given $\alpha_1(i,j) = 1$ and $\alpha_1(i,l) = 1$ the right-hand-side (RHS) of (6a) amounts to 1. In this case also the left-hand-side (LHS) of (6a) becomes 1 to fulfill the inequality (6a). Thus either $\alpha_1(j,l) = 1$ or $\alpha_2(j,l) = 1$. Reversely, assume that $\alpha_1(i,j) = 1$ and $\alpha_1(i,l) = 1$ does not hold, then the RHS of (6a) is less than 1, i.e., 0 or -1, while the LHS of (6a) is always greater than 0. Hence, constraint (6a) is fulfilled irrespectively of the choice of $\alpha_k(j,l)$, i.e., constraint (6a) enforces no logical implications.

Similarly for DAG \mathcal{D}_2 in Fig. 4, it is obvious that genes i,j belong to the hierarchical relationship class $k = 1$, i.e., $\alpha_1(i,j) = 1$. All genes $l \in \mathcal{G}_{\mathcal{D}_2} \setminus \{R\} : l > j > i$ which are in a linear pathway upstream of gene i , i.e. $\alpha_2(i,l) = 1$, must be located in SMAP \mathcal{S}_3 . Hence it directly follows from DAG \mathcal{D}_2 that also gene l must be in a linear pathway upstream of gene j , i.e., $\alpha_2(j,l) = 1$. This logical implication is compactly represented in constraint (6b). Under the assumption that $\alpha_1(i,j) = 1$ and $\alpha_2(i,l) = 1$, the RHS of (6b) amounts to 1 enforcing $\alpha_2(j,l) = 1$, so that the LHS of (6b) equals the RHS and the inequality in (6b) is fulfilled. Reversely, assume that $\alpha_2(i,l) = 0$, then the RHS of (6b) is less than 1 and

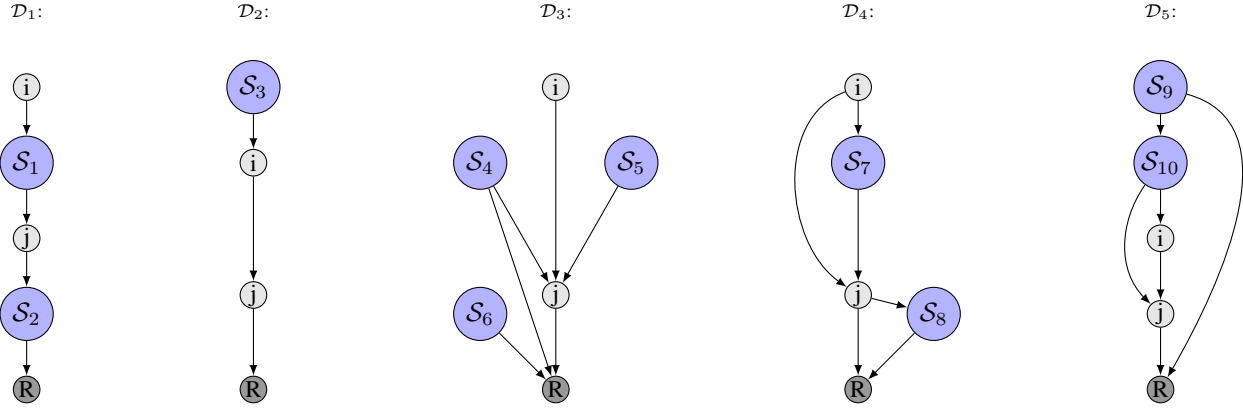


Fig. 4: Schematically reduced DAGs \mathcal{D}_1 to \mathcal{D}_5 corresponding to Eq. (6a)-Eq. (6e) respectively

hence the LHS of (6b) is always bigger than or equal to the RHS irrespectively of the choice of $\alpha_k(j, l)$, i.e., constraint (6a) enforces no logical implications. We can proceed in the same fashion to explain constraints (6c)-(6e) based on DAGs \mathcal{D}_3 to \mathcal{D}_5 as given in Fig. 4 respectively. Furthermore, with the same line of argument we can derive the sets \mathcal{L}_k for $k \in \mathcal{K} \setminus 1$ which reflect the logical implications among the selection variables under the assumptions that $\alpha_k(i, j) = 1$ for $k \in \mathcal{K} \setminus 1$. However, due to space limitations we omit the derivation of the full set of logical implications at this point and refer the interested reader to [32] where we will provide the full set of topology constraints \mathcal{L} as well as further supplementary material. The full set of topology constraints \mathcal{L} in (5d) can be computed as

$$\mathcal{L} = \bigcup_{k=1}^{|\mathcal{K}|} \{\mathcal{L}_k\}. \quad (7)$$

Finally, a considerable advantage of the presented algorithm is its ability to incorporate prior knowledge into the classification of the gene pairs $i, j \in \mathcal{G} : j > i$ to the most consistent hierarchical relationship classes. Assume that it is known from existing experimental results that two specific genes $i_0, j_0 \in \mathcal{G} : j_0 > i_0$ do not interact with each other. Then we can easily incorporate this prior knowledge into program O_{GENIE} in (5) by adding Eq. (8) as defined below

$$\alpha_3(i_0, j_0) = 1 \quad (8)$$

as a topology constraint to program O_{GENIE} . This property is very valuable since it allows the GENIE-algorithm to take advantage of existing results in genetic interactions research to improve the reliability of the classification.

B. Edge Computation

Based on the detected set of selection variables $A^{\text{O}_{\text{GENIE}}}$ which corresponds to the most consistent pattern of hierarchical relationship classes given the observed SK and DK phenotypes, an estimate $\mathcal{E}_{\text{GENIE}}$ of the true set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG \mathcal{D} can be computed. It can be theoretically proven that

the representation of an arbitrary DAG \mathcal{D} by its corresponding set of hierarchical relationship classes is not unique. $A^{\mathcal{D}}$ the set of selection variables which directly corresponds to the hierarchical relationship class pattern of DAG \mathcal{D} represents not only the true DAG \mathcal{D} , but also a set of similar DAGs which have minorly different sets of edges compared to the true DAG \mathcal{D} . Assume we are only given that $\alpha_4^{\mathcal{D}}(i, j) = 1$ for two arbitrary genes $i, j \in \mathcal{G} : j > i$ of DAG \mathcal{D} , then we suffer an information loss on the number of paths from gene i to the reporter node R which are independent of gene j . Similarly, given that $\alpha_5^{\mathcal{D}}(i, j) = 1$ for two arbitrary genes $i, j \in \mathcal{G} : j > i$ of DAG \mathcal{D} , we suffer an information loss on the number of paths from gene j to the reporter node R which are independent of gene i . Hence, this information loss yields ambiguities in computing the set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG \mathcal{D} based on the $A^{\mathcal{D}}$. In order to clarify the origin of the ambiguities further, let us turn to a simple example. Given

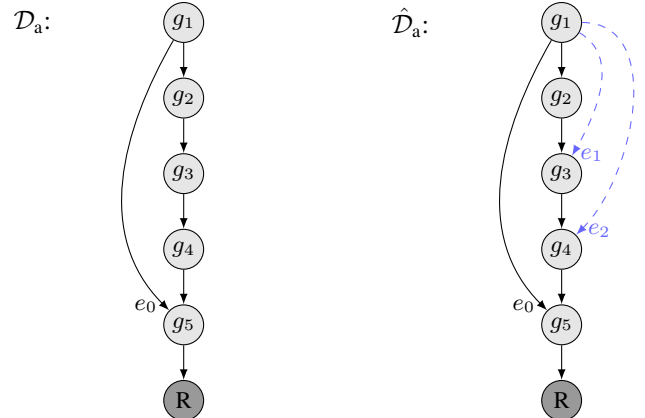


Fig. 5: Left: Original DAG \mathcal{D}_a with corresponding set of hierarchical relationship classes $A^{\mathcal{D}_a}$; Right: Reconstruction $\hat{\mathcal{D}}_a$ of DAG \mathcal{D}_a based on $A^{\mathcal{D}_a}$

DAG $\mathcal{D}_a = \{\mathcal{G}_{\mathcal{D}_a}, \mathcal{E}_{\mathcal{D}_a}\}$ as displayed on the LHS of Fig. 5 and the corresponding set of hierarchical relationship classes represented by the corresponding set of selection variables

$A^{\mathcal{D}_a}$. Note that $\alpha_4^{\mathcal{D}_a}(1, 2) = \alpha_4^{\mathcal{D}_a}(1, 3) = \alpha_4^{\mathcal{D}_a}(1, 4) = 1$ due to edge $e_0 \in \mathcal{E}_{\mathcal{D}_a}$ and $\alpha_1^{\mathcal{D}_a}(1, 5) = 1$. Now assume that we want to compute the topology of DAG \mathcal{D}_a , i.e., the set of edges $\mathcal{E}_{\mathcal{D}_a}$, based on $A^{\mathcal{D}_a}$. DAG \mathcal{D}_a displayed on the RHS of Fig. 5 shows the estimated topology of DAG \mathcal{D}_a based on $A^{\mathcal{D}_a}$. It can be shown that the black edges of DAG $\hat{\mathcal{D}}_a$ are necessary such that DAG $\hat{\mathcal{D}}_a$ is represented by $A^{\mathcal{D}_a}$. Edges e_1 and e_2 in DAG $\hat{\mathcal{D}}_a$ are optional in a sense that their existence has no effect on set $A^{\mathcal{D}_a}$. Edges e_1 and e_2 create two paths from gene g_1 to the reporter node R which are independent of gene g_2 and gene g_3 respectively. However, due to edge e_0 gene g_1 already has a path to the reporter node R which is independent of genes g_2 and g_3 . Since $\alpha_4^{\mathcal{D}_a}(1, 2) = \alpha_4^{\mathcal{D}_a}(1, 3) = \alpha_4^{\mathcal{D}_a}(1, 4) = 1$ do not contain information on the number of paths from gene g_1 to R that are independent of g_2 , g_3 and g_4 , edges e_1 and e_2 do not affect the pattern of hierarchical relationship classes representing DAG \mathcal{D}_a , i.e., $A^{\mathcal{D}_a}$ and hence this yields ambiguities in computing the topology of DAG \mathcal{D}_a based on its corresponding set of selection variables $A^{\mathcal{D}_a}$. Since it is a common assumption in genomics research, that GI-maps, i.e., DAGs, are not overly dense but rather sparse, we propose a policy which computes the sparsest DAG topology based on the detected pattern of hierarchical relationship classes. Given the detected pattern of hierarchical relationship classes of a DAG \mathcal{D} , i.e., $\hat{A}^{\mathcal{D}} = \bigcup_{\forall i, j \in \mathcal{G}: j > i} \{\hat{\alpha}_1^{\mathcal{D}}(i, j), \dots, \hat{\alpha}_5^{\mathcal{D}}(i, j)\}$,

we compute an estimate $\hat{\mathcal{E}}_{\mathcal{D}}$ of the true topology set $\mathcal{E}_{\mathcal{D}}$ of DAG \mathcal{D} according to the policy depicted in Tab. I where we make use of the symmetry properties $\hat{\alpha}_1^{\mathcal{D}}(i, j) = \hat{\alpha}_2^{\mathcal{D}}(j, i)$, $\hat{\alpha}_2^{\mathcal{D}}(i, j) = \hat{\alpha}_1^{\mathcal{D}}(j, i)$, $\hat{\alpha}_3^{\mathcal{D}}(i, j) = \hat{\alpha}_3^{\mathcal{D}}(j, i)$, $\hat{\alpha}_4^{\mathcal{D}}(i, j) = \hat{\alpha}_5^{\mathcal{D}}(j, i)$ and $\hat{\alpha}_5^{\mathcal{D}}(i, j) = \hat{\alpha}_4^{\mathcal{D}}(j, i)$ to redundantly expand the set of detected selection variables in order to obtain a compact formulation of the mutually exclusive conditions E_1 to E_4 as stated in Tab. I.

Assume that either condition E_1 or condition E_2 is fulfilled, then we conclude that there is an edge from gene i to gene j in DAG \mathcal{D} . Given that either condition E_3 or condition E_4 is fulfilled, we conclude that there exists an edge from gene j to gene i in DAG \mathcal{D} . We remark that there cannot be an edge between two genes $i, j \in \mathcal{G} : j > i$ if they are independent of each other, i.e., $\hat{\alpha}_3^{\mathcal{D}}(i, j) = 1$.

As described by E_1 there is an edge from gene i to gene j in DAG \mathcal{D} , if gene i is linearly upstream of gene j , i.e., $\hat{\alpha}_1^{\mathcal{D}}(i, j) = 1$, and there is no gene l in DAG \mathcal{D} that is linearly downstream of gene i , i.e., $\hat{\alpha}_1^{\mathcal{D}}(i, l) = 1$, and linearly upstream of gene j , i.e., $\hat{\alpha}_2^{\mathcal{D}}(j, l) = 1$. According to condition E_2 there is an edge from gene i to gene j in DAG \mathcal{D} , if gene i is upstream of gene j with at least one path from gene i to R which is independent of gene j , and furthermore there is no gene l in DAG \mathcal{D} that is either linearly downstream of gene i or downstream of gene i with gene i having at least one path to R that is independent of l , and, gene l is neither linearly upstream of gene j nor is gene l upstream of gene j with an independent path to R . In order to elucidate the effect of condition E_2 onto the edge computation, we briefly turn to DAG $\hat{\mathcal{D}}_a$ in Fig. 5. Condition E_2 ensures that the optional edges e_1 and e_2 are

Detection Policy: Compute sparsest DAG in line with $\hat{A}^{\mathcal{D}}$

E_1 :

$$\left(\hat{\alpha}_1^{\mathcal{D}}(i, j) = 1 \right) \wedge \left(\nexists l \in \mathcal{G} \setminus (i, j) : \right. \\ \left. \hat{\alpha}_1^{\mathcal{D}}(i, l) = 1 \wedge \hat{\alpha}_2^{\mathcal{D}}(j, l) = 1 \right)$$

• \implies there is an edge in DAG \mathcal{D} from gene i to gene j , i.e. $\{i, j\}$

E_2 :

$$\left(\hat{\alpha}_4^{\mathcal{D}}(i, j) = 1 \right) \wedge \left(\nexists l \in \mathcal{G} \setminus (i, j) : \right. \\ \left(\hat{\alpha}_1^{\mathcal{D}}(i, l) = 1 \vee \hat{\alpha}_4^{\mathcal{D}}(i, l) = 1 \right) \wedge \\ \left(\hat{\alpha}_2^{\mathcal{D}}(j, l) = 1 \vee \hat{\alpha}_5^{\mathcal{D}}(j, l) = 1 \right) \Big)$$

• \implies there is an edge in DAG \mathcal{D} from gene i to gene j , i.e. $\{i, j\}$

E_3 :

$$\left(\hat{\alpha}_2^{\mathcal{D}}(i, j) = 1 \right) \wedge \left(\nexists l \in \mathcal{G} \setminus (i, j) : \right. \\ \left. \hat{\alpha}_2^{\mathcal{D}}(i, l) = 1 \wedge \hat{\alpha}_1^{\mathcal{D}}(j, l) = 1 \right)$$

• \implies there is an edge in DAG \mathcal{D} from gene j to gene i , i.e. $\{j, i\}$

E_4 :

$$\left(\hat{\alpha}_5^{\mathcal{D}}(i, j) = 1 \right) \wedge \left(\nexists l \in \mathcal{G} \setminus (i, j) : \right. \\ \left(\hat{\alpha}_2^{\mathcal{D}}(i, l) = 1 \vee \hat{\alpha}_5^{\mathcal{D}}(i, l) = 1 \right) \wedge \\ \left(\hat{\alpha}_1^{\mathcal{D}}(j, l) = 1 \vee \hat{\alpha}_4^{\mathcal{D}}(j, l) = 1 \right) \Big)$$

• \implies there is an edge in DAG \mathcal{D} from gene j to gene i , i.e. $\{j, i\}$

TABLE I: Proposed sparse edge detection policy

not detected but only the necessary edges displayed in black color. We remark that conditions E_3 and E_4 can be elucidated by the same line of argument as used for conditions E_1 and E_2 , but due to space limitations we omit a detailed explanation at this point. Finally, we propose a condition from which all reporter node edges, i.e., all edges that connect gene $i \in \mathcal{G}$ with reporter node R in DAG \mathcal{D} can be computed. Based on the detected set of hierarchical relationship classes, i.e., $\hat{A}^{\mathcal{D}}$, we follow our policy of computing the necessary edges only. For clarity of presentation and notational compactness, we define set \mathcal{M}_i as

$$\mathcal{M}_i = \{l \in \mathcal{G} \setminus i \mid \hat{\alpha}_4^{\mathcal{D}}(i, l) = 1\}, \quad i = 1, \dots, G \quad (9)$$

containing all genes $l \in \mathcal{G}$ which are in class $k = 4$ with gene $i \in \mathcal{G}$, i.e., $\hat{\alpha}_4^{\mathcal{D}}(i, l) = 1$. Furthermore, we define set \mathcal{M}'_i as

$$\mathcal{M}'_i = \{l \in \mathcal{M}_i \mid \exists \tilde{l} \in \mathcal{M}_i \setminus l : \hat{\alpha}_3^{\mathcal{D}}(l, \tilde{l}) = 1\}, \quad i = 1, \dots, G \quad (10)$$

which contains all genes l of set \mathcal{M}_i that are independent of at least one other gene of set \mathcal{M}_i . Based on sets \mathcal{M}_i and \mathcal{M}'_i , we formulate condition E_R as stated in Tab. II. We conclude that there is an edge from gene i to reporter node R

$$E_R: \underbrace{\left(\nexists l : \hat{\alpha}_1^{\mathcal{D}}(i, l) = 1 \right)}_{=LHS} \wedge \underbrace{\left(\mathcal{M}_i = \emptyset \vee \mathcal{M}'_i = \emptyset \right)}_{=RHS}$$

TABLE II: Proposed reporter node edge detection policy

in DAG \mathcal{D} , if condition E_R is fulfilled. Given that gene i is linearly upstream of at least a single gene l , i.e., $\hat{\alpha}_1^{\mathcal{D}}(i, l) = 1$, there cannot exist an edge from gene i to reporter node R in DAG \mathcal{D} , since all paths from gene i to R pass through at least one other gene l . Conversely, if there is no such gene l that $\hat{\alpha}_1^{\mathcal{D}}(i, l) = 1$, then the LHS of E_R as given in Tab. II is fulfilled. The RHS of E_R accounts for our policy of detecting sparse DAGs only and is fulfilled if either \mathcal{M}_i , \mathcal{M}'_i or \mathcal{M}_i and \mathcal{M}'_i are empty. Note that given $\mathcal{M}_i = \emptyset$ it follows that $\mathcal{M}'_i = \emptyset$ as well, whereas the opposite is not true. In order to

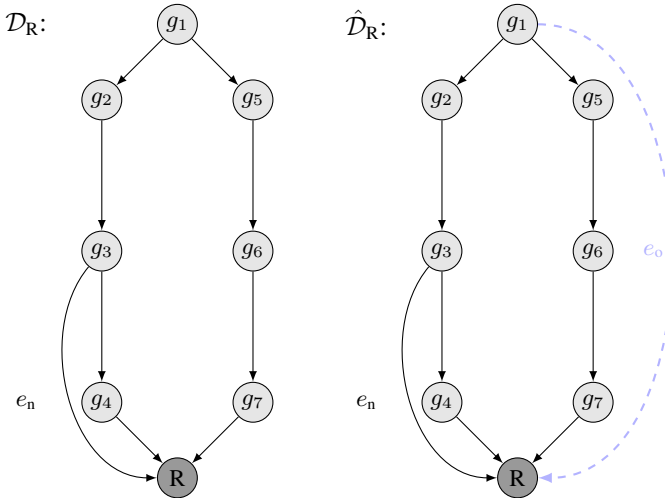


Fig. 6: Example DAG \mathcal{D}_R to elucidate the functionality of the RHS of condition E_R of Tab. II

explain the effect of the RHS of condition E_R in an intuitive manner, let us turn to DAG \mathcal{D}_R as displayed in Fig. 6. Assume we are given the pattern of hierarchical relationship classes that corresponds to DAG \mathcal{D}_R , i.e., $A^{\mathcal{D}_R}$ and we want to compute all reporter node edges based on $A^{\mathcal{D}_R}$, i.e., all edges that directly connect a gene in DAG \mathcal{D}_R with the reporter node R . Note that $\alpha_1^{\mathcal{D}_R}(2, 3) = 1$, $\alpha_5^{\mathcal{D}_R}(2, 4) = 1$, $\alpha_4^{\mathcal{D}_R}(3, 4) = 1$ and $\alpha_4^{\mathcal{D}_R}(1, l) = 1 \forall l \in \mathcal{G}_{\mathcal{D}_R} \setminus \{g_1, R\}$. It can be shown that gene g_3 fulfills the LHS of E_R , i.e., there is no gene which is linearly downstream of g_3 and furthermore $\mathcal{M}_3 = \{g_4\}$ and $\mathcal{M}'_3 = \emptyset$. Hence, condition E_R is fulfilled and edge e_n connecting g_3 and R is computed in DAG $\hat{\mathcal{D}}_R$ that is the reconstruction of DAG \mathcal{D}_R based on $A^{\mathcal{D}_R}$. Furthermore, for set $A^{\mathcal{D}_R}$ the edge e_n in the reconstructed DAG $\hat{\mathcal{D}}_R$ is necessary, since $\alpha_1^{\mathcal{D}_R}(2, 3) = 1$, $\alpha_5^{\mathcal{D}_R}(2, 4) = 1$ and

$\alpha_4^{\mathcal{D}_R}(3, 4) = 1$. In contrast to e_n edge e_o is not necessary for $A^{\mathcal{D}_R}$ to represent $\hat{\mathcal{D}}_R$, since $\alpha_4^{\mathcal{D}_R}(1, l) = 1 \forall l \in \mathcal{G}_{\mathcal{D}_R} \setminus \{g_1, R\}$ irrespectively of edge e_o . Hence, e_o is not detected, since $\mathcal{M}_1 \neq \emptyset$ and $\mathcal{M}'_1 \neq \emptyset$.

We obtain an estimate $\mathcal{E}_{\text{GENIE}}^{\hat{\mathcal{D}}}$ of the true set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG \mathcal{D} by setting $\hat{\mathcal{A}}^{\mathcal{D}} = A^{\text{OGENIE}}$ and evaluating conditions E_1 to E_4 and condition E_R as stated in Tab. I and Tab. II, respectively.

IV. GI-GENIE-ALGORITHM

In this section, we extend program OGENIE in (5) by incorporating GI-profile data, in order to jointly detect the most consistent pattern of hierarchical relationship classes and the corresponding set of edges \mathcal{E}_{GI} that is an estimate of the true graph topology of DAG \mathcal{D} , i.e., the set of edges $\mathcal{E}_{\mathcal{D}}$ which underlies the SK, DK and GI-profile data.

Let us define the following selection variables

$$\beta(i, j) = \begin{cases} 1 & \exists \text{ edge between } i, j \\ 0 & \text{no edge} \end{cases} \quad \forall i, j \in \mathcal{G} : j > i \quad (11)$$

where $\beta(i, j) = 1$ denotes that there is an edge between genes $i, j \in \mathcal{G} : j > i$ in DAG \mathcal{D} and $\beta(i, j) = 0$ denotes that there exists no edge between genes i and j . Note that unlike $\alpha_k(i, j) = 1$ for $k \in \mathcal{K}$, $\beta(i, j) = 1$ does not capture directionality information about the graph topology, i.e., $\beta(i, j) = 1$ states that there is an edge between genes i, j in DAG \mathcal{D} without specifying the hierarchy among both genes. The GI-GENIE program of jointly detecting the most consistent pattern of hierarchical relationship classes $\hat{\mathcal{A}}^{\text{OGENIE}}$ and the corresponding DAG topology \mathcal{E}_{GI} based on SK, DK and GI-profile data can be formulated as linear integer program

$\text{O}_{\text{GI-GENIE}} :$

$$\begin{aligned} \min_{\{\alpha_k(i, j), \beta(i, j), z_l(i, j)\}} & \lambda_d \sum_{i=1}^G \sum_{j=i+1}^G \left(\sum_{k=1}^{|\mathcal{K}|} s_k(i, j) \alpha_k(i, j) \right) \\ & - \lambda_s \sum_{i=1}^G \sum_{j=i+1}^G \left(\sum_l z_l(i, j) \right) \\ & - \lambda_c \sum_{i=1}^G \sum_{j=i+1}^G \rho(i, j) \beta(i, j) \\ & + \lambda_p \sum_{i=1}^G \sum_{j=i+1}^G \beta(i, j) \end{aligned} \quad (12a)$$

$$\text{s. t.:} \quad \text{Eq. (5b)- Eq. (5d)} \quad (12b)$$

$$\beta(i, j) \in \{0, 1\} \quad \forall i, j \in \mathcal{G} : j > i \quad (12c)$$

$$z_l(i, j) \in \{0, 1\} \quad \forall l \in \mathcal{G} \setminus \{i, j\},$$

$$\forall i, j \in \mathcal{G} : j > i \quad (12d)$$

$$1 - \alpha_3(i, j) \geq \beta(i, j)$$

$$\forall i, j \in \mathcal{G} : j > i \quad (12e)$$

$\mathcal{L}_c \implies$ additional topology constraints (12f)

$$|\mathcal{G}| - 2 + \beta(i, j) \geq 1 + \sum_{l \in \mathcal{G} \setminus \{i, j\}} z_l(i, j) \quad (12g)$$

$$\forall i, j \in \mathcal{G} : j > i$$

where $\rho(i, j)$ denotes a measure of the similarity between the GI-profiles of genes $i, j \in \mathcal{G} : j > i$ according to (1), the scalars $\lambda_d, \lambda_s, \lambda_c, \lambda_p$ are non-negative weighting constants and $z_l(i, j) \forall i, j, l \in \mathcal{G} : j > i, l \neq i, l \neq j$ are binary slack variables. Note that λ_s is assumed to be a very small non-negative constant, i.e. $0 \leq \lambda_s \ll 1$. The GI-profile (GIP) term

$$-\lambda_c \sum_{i=1}^G \sum_{j=i+1}^G \rho(i, j) \beta(i, j) + \lambda_p \sum_{i=1}^G \sum_{j=i+1}^G \beta(i, j) \quad (13)$$

in the objective function (12a) of program $\text{O}_{\text{GI-GENIE}}$ seeks to detect an edge between genes $i, j \in \mathcal{G} : j > i$ if the GI-profile similarity $\rho(i, j)$ is greater than the predefined threshold $\frac{\lambda_p}{\lambda_c}$. The slack term

$$-\lambda_s \sum_{i=1}^G \sum_{j=i+1}^G \left(\sum_l z_l(i, j) \right) \quad (14)$$

is necessary for the correct functionality of constraints (12f) and (12g) as explained below. The slack variables $z_l(i, j) \forall i, j, l \in \mathcal{G} : j > i, l \neq i, l \neq j$ are generally necessary to ensure that the information about the topology of DAG \mathcal{D} , which is encoded in the pattern of selection variables $\mathbf{A}^{\text{O}_{\text{GI-GENIE}}}$ detected by program $\text{O}_{\text{GI-GENIE}}$, is not contradicting with the set of edge selection variables $\{\hat{\beta}(i, j)\} \forall i, j \in \mathcal{G} : j > i$ detected by program $\text{O}_{\text{GI-GENIE}}$. In particular, given that the detected pattern of selection variables $\mathbf{A}^{\text{O}_{\text{GI-GENIE}}}$ enforces that there is an edge between genes i, j in DAG \mathcal{D} , then the slack variables ensure that the corresponding edge selection variable indicates that there is an edge between genes i, j , i.e., $\hat{\beta}(i, j) = 1$. Furthermore, given that the detected pattern of selection variables $\mathbf{A}^{\text{O}_{\text{GI-GENIE}}}$ enforces that there is no edge between genes i, j in DAG \mathcal{D} , then the slack variables ensure that the corresponding edge selection variable indicates that there is no edge between genes i, j , i.e., $\hat{\beta}(i, j) = 0$. On the contrary, assume that the detected edge selection variables enforce that there is an edge between genes i, j in DAG \mathcal{D} , i.e., $\hat{\beta}(i, j) = 1$, then the $z_l(i, j) \forall i, j, l \in \mathcal{G} : j > i, l \neq i, l \neq j$ ensure that the detected pattern of selection variables $\mathbf{A}^{\text{O}_{\text{GI-GENIE}}}$ must fulfill one of the conditions stated in Tab. I. Consequently, given that the detected edge selection variables enforce that there is no edge between genes i, j in DAG \mathcal{D} , i.e., $\hat{\beta}(i, j) = 0$, then the $z_l(i, j) \forall i, j, l \in \mathcal{G} : j > i, l \neq i, l \neq j$ ensure that the detected pattern of selection variables $\mathbf{A}^{\text{O}_{\text{GI-GENIE}}}$ does not fulfill any of the conditions stated in Tab. I.

Furthermore, let the auxiliary variables

$$q(i, j) = \begin{cases} 1 & \rho(i, j) \geq \frac{\lambda_c}{\lambda_p} \\ 0 & \rho(i, j) < \frac{\lambda_c}{\lambda_p} \end{cases} \quad \forall i, j \in \mathcal{G} : j > i \quad (15)$$

describe the detection of the edges of DAG \mathcal{D} based on GI-profile data only, where $q(i, j) = 1$ denotes that there is an edge between genes $i, j \in \mathcal{G} : j > i$ and $q(i, j) = 0$ denotes that there is no edge between genes $i, j \in \mathcal{G} : j > i$. Since any pattern of hierarchical relationship classes implies a specific set of edges and any set of edges implies a specific pattern of hierarchical relationship classes, there is a strong coupling of constraints, i.e. there are strong logical implications among the selection variables $\alpha_k(i, j) \forall i, j \in \mathcal{G} : j > i, \forall k \in \mathcal{K}$ and the selection variables $\beta(i, j) \forall i, j \in \mathcal{G} : j > i$, the constraints in Eq. (12e) to Eq. (12g) ensure that the detected hierarchical relationship classes and the corresponding edges, i.e., the $\alpha_k(i, j)$ and $\beta(i, j)$, are not mutually contradicting. Given that two genes $i, j \in \mathcal{G} : j > i$ in DAG \mathcal{D} are independent, i.e., $\alpha_3(i, j) = 1$, there cannot exist an edge between those genes in DAG \mathcal{D} , i.e., $\beta(i, j) = 0$. This logical implication is reflected by (12e). Set \mathcal{L}_c in (12f) and the linear integer inequalities in (12g) model conditions E_1 to E_4 of our proposed edge detection policy as stated in Tab. I. Since we do not want to redundantly expand the set of selection variables $\alpha_k(i, j)$ to all $i, j \in \mathcal{G} : j \neq i$ in order to not increase the complexity of program $\text{O}_{\text{GI-GENIE}}$, we have to consider three cases when formulating conditions E_1 to E_4 of Tab. I as linear integer inequalities, i.e., $i, j, l \in \mathcal{G} : l > j > i$, $i, j, l \in \mathcal{G} : j > i > l$ and $i, j, l \in \mathcal{G} : j > l > i$. Then the constraints in set $\mathcal{L}_{c,1}$

$$\mathcal{L}_{c,1} = \begin{cases} 1 - \beta(i, j) \geq \alpha_1(i, j) + \alpha_1(i, l) + \alpha_2(j, l) - 2 & (16a) \end{cases}$$

$$1 - z_l(i, j) \geq \alpha_1(i, j) + \alpha_1(i, l) + \alpha_2(j, l) - 2 \quad (16b)$$

$$\frac{1}{2} (\alpha_1(i, l) + \alpha_2(j, l)) \geq \alpha_1(i, j) - z_l(i, j) \quad (16c)$$

$$1 - \beta(i, j) \geq \alpha_2(i, j) + \alpha_2(i, l) + \alpha_1(j, l) - 2$$

$$1 - z_l(i, j) \geq \alpha_2(i, j) + \alpha_2(i, l) + \alpha_1(j, l) - 2$$

$$\frac{1}{2} (\alpha_2(i, l) + \alpha_1(j, l)) \geq \alpha_2(i, j) - z_l(i, j) \quad (16d)$$

$$1 - \beta(i, j) + q(i, j) \geq \alpha_4(i, j) + \alpha_1(i, l) + \alpha_4(i, l) + \alpha_2(j, l) + \alpha_5(j, l) - 2 \quad (16e)$$

$$1 - z_l(i, j) + q(i, j) \geq \alpha_4(i, j) + \alpha_1(i, l) + \alpha_4(i, l) + \alpha_2(j, l) + \alpha_5(j, l) - 2 \quad (16f)$$

$$\frac{1}{2} (\alpha_1(i, l) + \alpha_4(i, l) + \alpha_2(j, l) + \alpha_5(j, l)) \geq \alpha_4(i, j) - z_l(i, j) - q(i, j) \quad (16g)$$

$$2 - \beta(i, j) - q(i, j) \geq \alpha_4(i, j) + \alpha_1(i, l) + \alpha_2(j, l) + \alpha_5(j, l) - 2 \quad (16h)$$

$$2 - z_l(i, j) - q(i, j) \geq \alpha_4(i, j) + \alpha_1(i, l)$$

$$\begin{aligned}
& + \alpha_2(j, l) + \alpha_5(j, l) - 2 \\
& \frac{1}{2} (\alpha_1(i, l) + \alpha_2(j, l)) \geq \\
& \alpha_4(i, j) - z_l(i, j) - 1 + q(i, j)
\end{aligned} \tag{16i}$$

$$\begin{aligned}
& 1 - \beta(i, j) + q(i, j) \geq \alpha_5(i, j) + \alpha_2(i, l) + \\
& \alpha_5(i, l) + \alpha_1(j, l) + \alpha_4(j, l) - 2 \\
& 1 - z_l(i, j) + q(i, j) \geq \alpha_5(i, j) + \alpha_2(i, l) + \\
& \alpha_5(i, l) + \alpha_1(j, l) + \alpha_4(j, l) - 2 \\
& \frac{1}{2} (\alpha_2(i, l) + \alpha_5(i, l) + \alpha_1(j, l) + \alpha_4(j, l)) \geq \\
& \alpha_5(i, j) - z_l(i, j) - q(i, j)
\end{aligned} \tag{16k}$$

$$\begin{aligned}
& 2 - \beta(i, j) - q(i, j) \geq \alpha_5(i, j) + \alpha_2(i, l) \\
& + \alpha_5(i, l) + \alpha_1(j, l) - 2 \\
& 2 - z_l(i, j) - q(i, j) \geq \alpha_5(i, j) + \alpha_2(i, l) \\
& + \alpha_5(i, l) + \alpha_1(j, l) - 2 \\
& \frac{1}{2} (\alpha_2(i, l) + \alpha_1(j, l)) \geq \\
& \alpha_5(i, j) - z_l(i, j) - 1 + q(i, j) \\
& \left. \right\} \forall i, j, l \in \mathcal{G} : l > j > i
\end{aligned} \tag{16l}$$

model the logical implications among the selection variables $\alpha_k(i, j), \alpha_{k'}(i, l), \alpha_{k''}(j, l)$ and $\beta(i, j)$ for $k, k', k'' \in \mathcal{K}$, $\forall i, j, l \in \mathcal{G} : l > j > i$. Together with (12g) and the slack term in (14), constraints (16a)-(16c) model condition E_1 of our detection policy taking into account the GI-profile similarity information $\rho(i, j)$ via selection variables $\beta(i, j)$. Assume that based on the SK and DK phenotypes it is most consistent that $\alpha_1(i, j) = \alpha_1(i, l) = \alpha_2(j, l) = 1$ for at least one gene l in DAG \mathcal{D} which corresponds to condition E_1 being violated. Hence, there cannot exist an edge between genes i and j in DAG \mathcal{D} . In this case the RHS of (16a) amounts to 1 which enforces the LHS of (16a) to amount to 1 as well, i.e., $\beta(i, j) = 0$. Note that for $\alpha_1(i, j) = \alpha_1(i, l) = \alpha_2(j, l) = 1$, (16c) makes no restrictions on $z_l(i, j)$ but by (16b) $z_l(i, j)$ is forced to 0, thus (12g) is fulfilled as well. Furthermore, assume for genes $i, j \in \mathcal{G} : j > i$, based on the SK and DK phenotypes, it is most consistent that $\alpha_1(i, j) = 1$, but $\alpha_1(i, l)$ and $\alpha_2(j, l)$ are not jointly 1 for all other genes $l \in \mathcal{G} : l > j > i$, i.e., $\alpha_1(i, l) + \alpha_1(j, l) < 2$, then there is an edge between genes i, j in DAG \mathcal{D} according to condition E_1 . In this case it is obvious that (16a) and (16b) are always fulfilled, i.e., there are no restrictions on $\beta(i, j)$ and $z_l(i, j)$ by (16a) and (16b). However, due to the slack term in (14), it follows that $z_l(i, j) = 1 \forall l \in \mathcal{G} : l > j > i$ and $\beta(i, j)$ is set to 1 in order to fulfill (12g).

Given that the GI-profile similarity data strongly supports that there is no edge between genes i, j in DAG \mathcal{D} , i.e., $\beta(i, j) = 0$, and $\alpha_1(i, j) = 1$ is most consistent based on the SK and DK phenotypes measured, then it follows from (12g) that there must be at least one $l \in \mathcal{G} : l > j > i$ for which

$z_l(i, j) = 0$. In this case, with $\beta(i, j) = 0$, $\alpha_1(i, j) = 1$ and $z_l(i, j) = 0$, the RHS of (16c) amounts to 1, forcing the LHS of (16c) to amount to 1 as well, i.e., $\alpha_1(i, l) = 1$ and $\alpha_2(j, l) = 1$, which is together with the assumption of $\alpha_1(i, j) = 1$ a combination that violates the existence of a direct edge between genes i and j . Furthermore, note that (16a) and (16b) do not have any implications on the selection variables $\alpha_1(i, j), \alpha_1(i, l), \alpha_2(j, l)$ for the case that $\beta(i, j) = 0$ and $z_l(i, j) = 0$.

Assume that the GI-profile similarity data strongly supports that there is an edge between genes i, j in DAG \mathcal{D} , i.e., $\beta(i, j) = 1$, and $\alpha_1(i, j) = 1$ is most consistent based on the SK and DK phenotypes measured, then according to (16a) there cannot be any gene $l \in \mathcal{G} : l > j > i$ for which $\alpha_1(i, l) = 1$ and $\alpha_2(j, l) = 1$. Hence, the RHS of (16b) is smaller or equal to 0 which enforces $z_l(i, j) = 1 \forall l \in \mathcal{G} : l > j > i$ due to the slack term in (14). Thus, (12g) is fulfilled with equality. Note that in this case (16c) is always fulfilled. Together with (12g) and the slack term in (14), the three inequalities in (16d) model condition E_2 where we can elucidate their functionality in the same fashion as before. Constraints (16e) to (16j) along with (12g) and the slack term in (14) model a minor modification of condition E_3 where we not only detect all necessary edges, but also optional edges given that their existence is strongly supported by the GI-profile. Given that the existence of an edge between genes $i, j \in \mathcal{G} : j > i$ in DAG \mathcal{D} is not strongly supported by the GI-profile, i.e. $q(i, j) = 0$, constraints (16e) to (16g) along with (12g) and (14) model condition E_3 which only allows necessary edges to be detected and we can elucidate their functionality in the same fashion as in (16a) to (16c). Note that (16h) to (16j) are always fulfilled for $q(i, j) = 0$, i.e., no implications among the selection variables $\alpha_k(i, j)$ and $\beta(i, j)$ are posed. Assuming that the existence of an edge between genes $i, j \in \mathcal{G} : j > i$ in DAG \mathcal{D} is strongly supported by the GI-profile, i.e. $q(i, j) = 1$, then the constraints in (16e) and (16g) are always fulfilled, i.e., no implications among the selection variables $\alpha_k(i, j)$ and $\beta(i, j)$ are posed by (16e) and (16g). However, constraints (16h) and (16j) pose relaxed logical implications among the selection variables $\alpha_k(i, j)$ and $\beta(i, j)$ compared to constraints (16e) to (16g). Hence given that $q(i, j) = 1$ and $\alpha_4(i, j) = 1$, an edge between genes $i, j \in \mathcal{G} : j > i$ in DAG \mathcal{D} is detected if it is allowed by the pattern of hierarchical relationship classes. Constraints (16k) to (16l) along with (12g) and the slack term in (14) model a minor modification of condition E_4 where we not only detect all necessary edges, but also optional edges given that their existence is strongly supported by the GI-profile. Furthermore, the functionality of constraints (16k) to (16l) can be explained with the same line of argument as used to elucidate constraints (16e) to (16j).

Denote $\mathcal{L}_{c,2}$ and $\mathcal{L}_{c,3}$ as the sets of topology constraints that model the logical coupling among the selection variables $\alpha_k(i, j), \alpha_{k'}(i, l), \alpha_{k''}(j, l)$ and $\beta(i, j)$ for $k, k', k'' \in \mathcal{K}$ and $i, j, l \in \mathcal{G} : j > i > l$ and $i, j, l \in \mathcal{G} : j > l > i$ respectively. Then the full set of coupled constraints of the

selection variables $\alpha_k(i, j)$ and $\beta(i, j)$ is given by

$$\mathcal{L}_c = \bigcup_{\kappa=1}^3 \{\mathcal{L}_{c,\kappa}\} \quad (17)$$

where we again refer the interested reader to [32] for a detailed description of \mathcal{L}_c . We obtain an estimate \mathcal{E}_{GI} of the true set of edges $\mathcal{E}_{\mathcal{D}}$ of DAG \mathcal{D} based on the computed set of edge selection variables $\hat{\beta}(i, j)$ of program $O_{GI-GENIE}$ where we infer the directionality of the edges according to $A^{O_{GI-GENIE}}$ that is the most consistent pattern of hierarchical relationship classes given the observed SK, DK and GI-profile data. Note that all reporter node edges are computed according to our proposed reporter node edge detection policy as given in Tab. II. Since the reporter node is an artificial node in the concept of a DAG, there is no GI-profile similarity data $\rho(i, R) \forall i \in \mathcal{G}$ and thus, no edge selection variable $\beta(i, R) \forall i \in \mathcal{G}$ according to 11.

V. SEQUENTIAL SCALABILITY TECHNIQUE

Due to the combinatorial nature of problems O^{GENIE} and $O^{GI-GENIE}$, the GENIE algorithm and GI-GENIE algorithm, respectively, cannot be applied to the data of large sets of genes \mathcal{G} , since the complexity of problems O^{GENIE} and $O^{GI-GENIE}$ grows exponentially with the number of genes and both problems become NP-hard. In order to nevertheless obtain statistically stressable statements about the interactions among genes in a large set of genes \mathcal{G} , we propose the sequential-scalability (SEQSCA)-technique which is based on the GENIE-algorithm and the GI-GENIE algorithm, respectively.

In particular, we create a sequence of S subsets $\{\mathcal{G}_s\}_1^S$ of the full set of genes \mathcal{G} , i.e., $\mathcal{G}_s \subset \mathcal{G}$, $\forall s \in \{1, \dots, S\}$, where we estimate the topology $\mathcal{E}_{\mathcal{D},s}$ of each DAG \mathcal{D}_s , underlying the data of the subset of genes \mathcal{G}_s , by the GENIE or GI-GENIE-algorithm, respectively, in order to translate the estimated topology $\mathcal{E}_{\mathcal{D},s}$ of DAG \mathcal{D}_s into the corresponding adjacency matrix M_s for each $s \in \{1, \dots, S\}$. Based on the computed sequence of adjacency matrices $\{M_s\}_1^S$, we iteratively compute the reliability matrix $M \in [0, 1]^{N \times N}$ of the full set of genes \mathcal{G} in such a way, that each entry $[M]_{i,j \in \mathcal{G}}$ describes the empirical probability of an edge to exist between genes $i, j \in \mathcal{G}$, i.e., the empirical probability that genes $i, j \in \mathcal{G}$ directly interact with each other, where a value of 0 means that there is an interaction between the considered pair of genes with probability 0 and a value of 1 means that the considered pair of genes interacts with probability 1.

In particular, in each iteration s we consider a subset \mathcal{G}_s of size $N_S \ll |\mathcal{G}|$ of the full set of genes \mathcal{G} , where each gene of \mathcal{G}_s is selected from \mathcal{G} without replacement with equal probability. Based on the selected subset \mathcal{G}_s , we compute in each iteration s an estimate $\mathcal{E}_{\mathcal{D},s}$ of the true topology of DAG \mathcal{D}_s , underlying the observed data of the genes in subset \mathcal{G}_s , by the GENIE or the GI-GENIE-algorithm, respectively. Furthermore, the topology estimate $\mathcal{E}_{\mathcal{D},s}$ of DAG \mathcal{D}_s is translated into the corresponding adjacency matrix M_s . The update of the reliability matrix for iteration s is computed according to

Eq. (18)

$$[M^{(s+1)}]_{i,j} = [M^{(s)}]_{i,j} + [M_s]_{\kappa_i, \kappa_j} \quad \forall i, j \in \mathcal{G}_s \quad (18)$$

with $M^{(s)}$ being the $N \times N$ reliability matrix at iteration s , $\kappa_i \in \{1, \dots, N_S\} \forall i \in \mathcal{G}_s$, $\cup_i \kappa_i = \{1, \dots, N_S\}$ and $\kappa_i < \kappa_j$ for all $i < j$. Finally, we obtain the reliability matrix M of the full set of genes \mathcal{G} by normalizing each entry $[M^{(S)}]_{i,j}$ by $n_{i,j}$ that is the frequency of how often detecting an edge between genes i and j has been considered.

Initialization: $M^{(0)} = \mathbf{0}_{N \times N}$; $M_{s=0} = \mathbf{0}_{N_S \times N_S}$; frequency counter $n_{i,j}^{(0)} = 0$

Repeat:

- 1 : Select subset \mathcal{G}_s of size N_S from \mathcal{G} ; draw each gene from \mathcal{G} with equal probability without replacement
- 2 : Update: $n_{i,j}^{(s+1)} = n_{i,j}^{(s)} + 1$ for all $i, j \in \mathcal{G}_s$
- 3 : Estimate the DAG topology \mathcal{D}_s of set \mathcal{G}_s using GENIE, GI-GENIE, respectively; $\Rightarrow M_s$
- 4 : Update reliability matrix $M^{(s)}$ according to Eq. (18)
- 7 : Update iteration number: $s \leftarrow s + 1$

Until: $s = S$;
Set $[M]_{i,j} = [M^{(S)}]_{i,j} / n_{i,j}^{(S)} \forall i, j \in \mathcal{G}$

TABLE III: Summary of the proposed SEQSCA-algorithm

In Tab. III, we have summarized the SEQSCA-technique. Finally, by means of the SEQSCA-technique, we are able to provide statistically stressable statements about the interactions among the genes from a large set \mathcal{G} by using the GENIE or GI-GENIE algorithm, respectively, in a sequential fashion.

VI. SIMULATION RESULTS

In this section, we first demonstrate the performance of the GENIE-algorithm and the GI-GENIE-algorithm with respect to conventional techniques for simulated data and further provide statistically stressable statements on the interactions among the genes from a large set of genes based on real data using the proposed SEQSCA-technique.

A. Synthetic Data Results

We have generated the ideal SK phenotypes $R(i) \in \mathbb{R}$ for all $i \in \mathcal{G}$ as well as the ideal DK phenotypes $R(i, j) \in \mathbb{R}$ for all $i, j \in \mathcal{G} : j > i$ according to the model of [7] as displayed in Fig. 2, where we have corrupted the ideal SK and DK phenotypes by independently and identically distributed zero-mean Gaussian noise with variance σ^2 . Furthermore, the GI-profile similarity data $\rho(i, j) \forall i, j \in \mathcal{G} : j > i$ has been generated on the basis of the SK and DK phenotypes. We compare both the GENIE-algorithm and the GI-GENIE-algorithm with the well known GI-profile similarity approach ([28],[7]), where the Pearson correlation between the GI-profiles of genes i and j is computed and an edge in the DAG is detected if the Pearson correlation is above a pre-defined threshold

t_{corr} , where the directionality is inferred from the selection variable $\alpha_k(i, j)$ corresponding to the least mismatch model $\mu_k(i, j)$. Furthermore, we compare our proposed methods with the solution of program O_{GENIE} without considering set \mathcal{L} as a constraint, which means simply classifying each pair i, j to the least mismatch scoring hierarchical relationship class based on the SK and DK phenotypes $R(i)$ and $R(i, j)$ respectively, without ensuring that the resulting pattern of hierarchical relationship classes represents a valid DAG.

For the simulations, we consider a total of 10 genes in order to limit the Monte-Carlo simulation time. In Fig. 7 we display the false detection percentage of edges P_{ed} in the detected DAG normalized to the true number of edges $|\mathcal{E}_{\mathcal{D}}|$ as defined in Eq. (19) versus the SNR.

$$P_{\text{ed}} = \frac{|\left(\mathcal{E}_{\mathcal{D}} \cup \hat{\mathcal{E}}_{\mathcal{D}}\right) \setminus \mathcal{E}_{\mathcal{D}}|}{|\mathcal{E}_{\mathcal{D}}|} \quad (19)$$

In Fig. 8 we display the percentage of undetected edges P_{mis} in the detected DAG normalized to the true number of edges $|\mathcal{E}_{\mathcal{D}}|$ as defined in Eq. (20) versus the SNR, i.e.,

$$P_{\text{mis}} = \frac{|\left(\mathcal{E}_{\mathcal{D}} \cup \hat{\mathcal{E}}_{\mathcal{D}}\right) \setminus \hat{\mathcal{E}}_{\mathcal{D}}|}{|\mathcal{E}_{\mathcal{D}}|} \quad (20)$$

In Fig. 7 we observe that in the low SNR regime, the Pearson correlation based method performs best in terms of false detection percentage of edges P_{ed} , however it fails to improve performance with increasing SNR, because for correct directionality information of the edges this approach relies on the hierarchical relationship classes detected by method O_{GENIE} without considering \mathcal{L} . Especially in the high SNR regime, the proposed GENIE and GI-GENIE methods clearly outperform program O_{GENIE} without the topology rule-set \mathcal{L} and approach and respectively reach the performance of the Pearson correlation method. However, the very good performance of the Pearson correlation method in terms of false detection percentage of edges P_{ed} according to Eq. (19) comes at the cost of a rather poor performance in terms of the percentage of undetected edges P_{mis} according to Eq. (20) as can be seen in Fig. 8. In particular, in terms of the percentage of undetected edges P_{mis} all of the proposed methods outperform the Pearson correlation method. Note that in the high SNR regime, the GI-GENIE of combining SK, DK and GI-profile similarity data yields the best of both worlds, i.e. it shows an equivalent performance as the Pearson correlation method in terms of false detection percentage of edges P_{ed} , as well as an improvement of the strong performance of the GENIE method in terms of the percentage of undetected edges P_{mis} .

B. Real Data Results

In this section, we apply the above mentioned SEQSCA-algorithm to the dataset of [29], in order to yield statistically stressable statements about the interactions among the genes considered in [29]. In [29], the organism under study is yeast and the cell process measured is the colony size as a proxy of

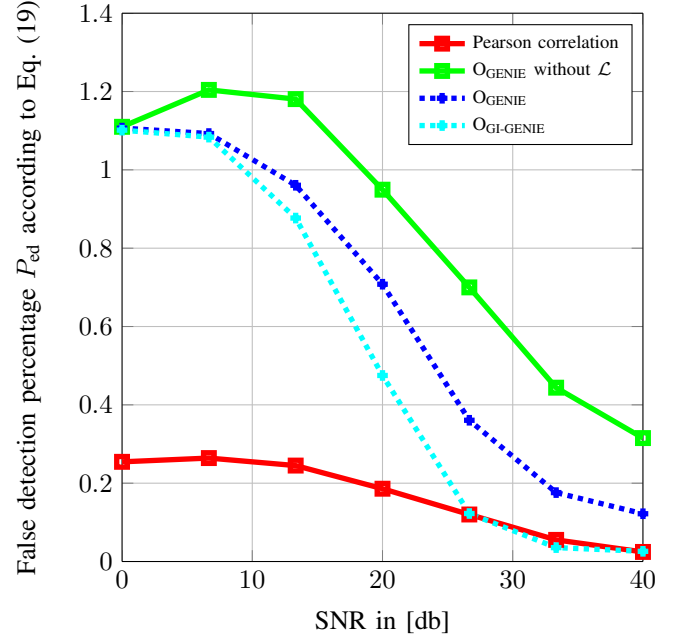


Fig. 7: D_{ed} versus SNR; $t_{\text{corr}} = .6$; 200 Monte Carlo runs; $\lambda_d = 1$, $\lambda_s = 8e - 6$, $\lambda_c = 20$, $\lambda_p = 16$

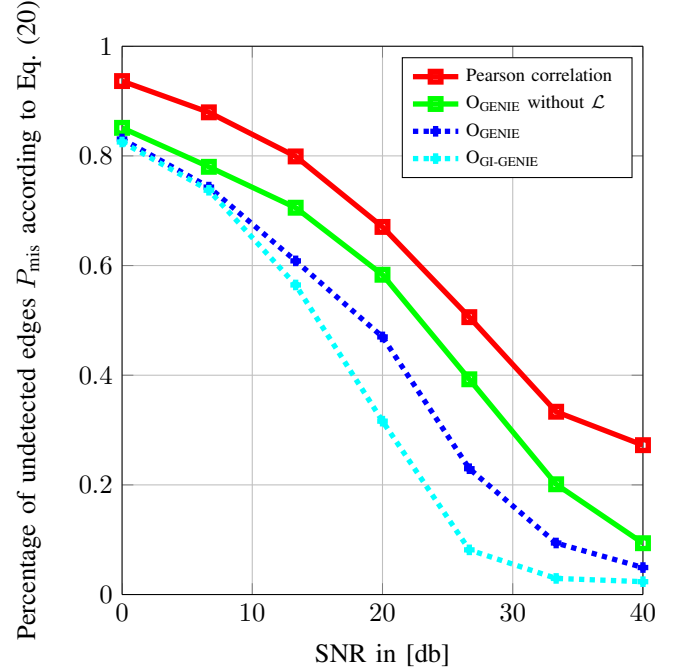


Fig. 8: D_{mis} versus SNR; $t_{\text{corr}} = .6$; 200 Monte Carlo runs; $\lambda_d = 1$, $\lambda_s = 8e - 6$, $\lambda_c = 20$, $\lambda_p = 16$

fitness. For the sake of computation time, we restrict the full set of genes under study in [29], i.e., \mathcal{G} , to contain the first 200 genes of the *query list* of [29]. Furthermore, the length of the sequence of subsets S is set to 50000 and the subset size N_S of each selected set \mathcal{G}_s in the SEQSCA-algorithm is set to 10.

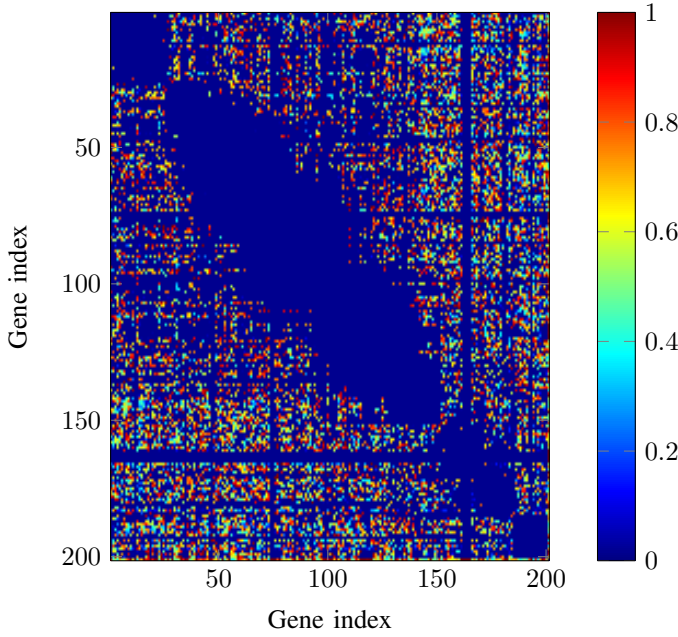


Fig. 9: Reliability matrix M for the SEQSCA-technique using GENIE-algorithm; $S = 50000$ subsets considered; subset size $N_S = 10$

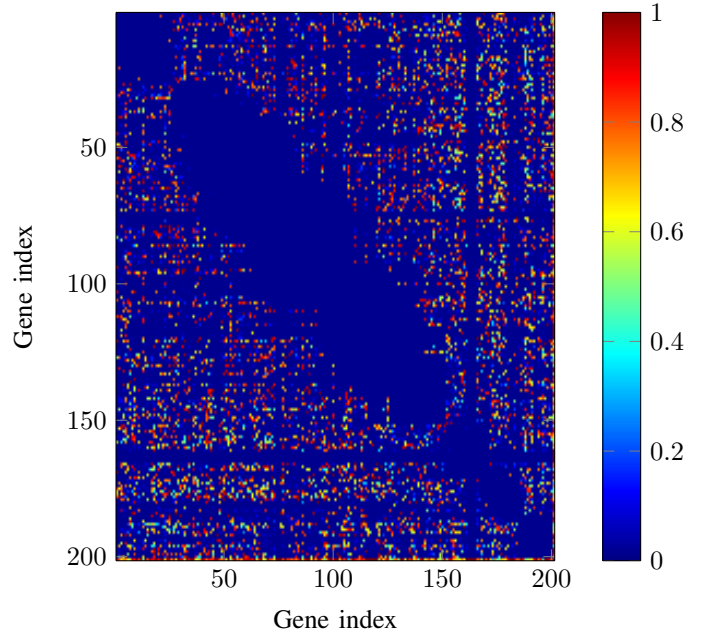


Fig. 10: Reliability matrix M for the SEQSCA-technique using GI-GENIE-algorithm; $S = 50000$ subsets considered; subset size $N_S = 10$; $\lambda_d = 1e3$, $\lambda_s = 8e - 6$, $\lambda_c = 1$, $\lambda_p = .85$

In Fig. 9 we show the reliability matrix $M \in [0, 1]^{200 \times 200}$ for $S = 50000$ subsets, where we have applied the SEQSCA-technique with the GENIE-algorithm for estimating in each iteration s the topology of DAG \mathcal{D}_s underlying the data of subset \mathcal{G}_s . It can be observed that only for a few pairs of genes $i, j \in \mathcal{G}$ edges have been detected with high empirical probability, which is in line with the common assumption in genomics research that genetic interactions are generally sparse. Furthermore, in Fig. 9 the empirical probability of an edge to be detected is either smaller than 30% or bigger than 70% for 87% of the gene pairs $i, j \in \mathcal{G}$. Hence, for most gene pairs the proposed SEQSCA-technique, using the GENIE-algorithm, is able to provide a statistically stressable statement whether a two genes i, j interact with each other or not. In Fig. 9 we show the reliability matrix $M \in [0, 1]^{200 \times 200}$ for again $S = 50000$ subsets, where we have applied the SEQSCA-technique with the GI-GENIE-algorithm for estimating in each iteration s the topology of DAG \mathcal{D}_s underlying the data of subset \mathcal{G}_s . In this case, the GI-profile data $\rho(i, j) \forall i, j \in \mathcal{G} : i \neq j$ has been computed based on the SK and DK phenotypes of [29] according to Eq. (1). In Fig. 9, the empirical probability that two genes interact with each other, is either smaller than 30% or bigger than 70% for 95% of the gene pairs $i, j \in \mathcal{G}$. Thus, the incorporation of GI-profile data into the topology estimation process in each iteration s of the SEQSCA-technique enhances the statistical reliability of the statements made by the SEQSCA-technique, i.e., whether two genes i, j interact directly interact with each other or not.

VII. CONCLUSION

In this paper we have considered the problem of learning the interactions between genes in a genetic network. Our approach for computing the interactions among the genes, i.e., the genetic-interaction map which is essentially a DAG, uses the interaction models of [7] and combines it with GI-profile data, that can be computed based on the Pearson correlation of the SK and DK phenotypes of the set of genes under study, or it can be extracted from some data base where other types of GI-profile data are stored. Furthermore, we have proposed an scalability approach, called SEQSCA, which uses the proposed GENIE and GI-GENIE-methods to yield statistically reliable statements, whether two genes out of large set of genes interact with each other or not.

REFERENCES

- [1] B. Snijder, P. Liberali, M. Frechin, T. Stoeger and L. Pelkmans, *Predicting functional gene interactions with the hierarchical interaction score*, Nature Methods, VOL.10 NO.11, November 2013
- [2] A. Baryshinkova et al. *Quantitative analysis of fitness and genetic interactions in yeast on a genome scale*, Nature Methods, December 2010
- [3] S.R. Collins, A. Roguev, N.J. Krogan *Quantitative Genetic Interaction Mapping Using the E-MAP Approach*, Methods Enzymol., 2010
- [4] R.O. Linden, V.P. Eronen, T. Aittokallio *Quantitative maps of genetic interactions in yeast - Comparative evaluation and integrative analysis*, BMC Systems Biology, 2011
- [5] S.J. Dixon, M. Constanzo, A. Baryshinkova, B. Andrews, C. Boone *Systematic Mapping of Genetic Interaction Networks*, The Annual Review of Genetics, 2009
- [6] G.N. Brock et al. *Methods for detecting gene gene interaction in multiplex extended pedigrees*, BMC Genetics, 2005

- [7] A. Battle, M.C. Jonikas, P. Walter, J.S. Weissman and D. Koller *Automated identification of pathways from quantitative genetic interaction data*, Molecular Systems Biology 6, 2010
- [8] T.C. Hu, A.B. Kahng *Linear and Integer Programming in Practice*, Springer International Publishing, Schweiz, 2016, ISBN-10: 3319239996
- [9] G. Sierksma *Linear and Integer Programming: Theory and Practice, Second Edition*, CRC Press, Boca Raton FL 33487-2741, 2001, ISBN-10: 0824706730
- [10] G. Sierksma, Y. Zwols *Linear and Integer Optimization: Theory and Practice, Third Edition*, CRC Press, Boca Raton FL 33487-2741, 2015, ISBN-10: 1498710166
- [11] E. Demirel, N. Demirel, H. Gökçen *A mixed integer linear programming model to optimize reverse logistics activities of end-of-life vehicles in Turkey*, Journal of Cleaner Production, 2016
- [12] C.H. Antunes, M.J. Alves, J. Climaco *Multiobjective Linear and Integer Programming*, Springer International Publishing, Schweiz, 2016, ISBN-13: 9783319287447
- [13] M. Diaby, M.H. Karwan *Advances in Combinatorial Optimization*, World Scientific Publishing Co. Pte. Ltd., Singapore, 2016, ISBN-10: 9814704873
- [14] A.H.Y. Tong et al. *Systematic genetic analysis with ordered arrays of yeast deletion mutants*, Science 294, 2001
- [15] R. Diestel *Graphentheorie*, Springer-Verlag, Heidelberg, 2012, ISBN 978-3-642-14911-5
- [16] C.H. Papadimitriou, K. Steiglitz *Combinatorial optimization: algorithms and complexity*, Mineola NY, 1998, ISBN 0486402584
- [17] A. Jaimovich et al. *Modularity and directionality in genetic interaction maps*, Nature Methods, 2010, Vol. 26
- [18] A. Baryshnikova, M. Constanzo, C.L. Myers, B. Andrews, C. Boone *Genetic Interaction Networks: Toward an Understanding of Heritability*, Annual Review of Genomics and Human Genetics, 2013
- [19] A. Rogueav et al. *Quantitative genetic-interaction mapping in mammalian cells*, Nature Methods, February 2013, Vol. 10
- [20] M. Constanzo et al. *The genetic landscape of a cell*, Science 327:425-431, 2010
- [21] V. Balakrishnan, S. Boyd, S. Balemi *Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems*, International Journal of Robust and Nonlinear Control, 1(4):295317, October/December 1991
- [22] E.L. Lawler, D.E. Wood *Branch-and-bound methods: A survey*, Operations Research, 14:699719, 1966
- [23] R.E. Moore *Global optimization to prescribed accuracy*, Computers and Mathematics with Applications, 21(6/7):2539, 1991
- [24] Y. Cheng, M. Pesavento *Joint Rate Adaptation and Downlink Beamforming Using Mixed Integer Conic Programming*, IEEE Transactions on Signal Processing, 2013
- [25] Y. Cheng, M. Pesavento *An Optimal Iterative Algorithm for Codebook-Based Downlink Beamforming*, IEEE Signal Processing Letters, 2013
- [26] Y. Cheng, M. Pesavento *Joint Optimization of Source Power Allocation and Distributed Relay Beamforming in Multiuser Peer-to-Peer Relay Networks*, IEEE Transactions on Signal Processing, Vol. 60, No 6, pp. 2395-2404, 2012
- [27] Y. Cheng, M. Pesavento, A. Philipp *Joint Network Optimization and Downlink Beamforming for CoMP Transmissions using Mixed Integer Conic Programming*, IEEE Transactions on Signal Processing, Vol. 61, 2013
- [28] M. Babu et al. *Quantitative Genome-Wide Genetic Interaction Screens Reveal Global Epistatic Relationships of Protein Complexes in Escherichia coli*, PLOS Genetics, February 2014, Volume 10
- [29] M. Costanzo et al. *DRYGIN - Data Repository of Yeast Genetic Interactions*, Terence Donnelly Centre for Cellular and Biochemical Research, University of Toronto, <http://drygin.ccbr.utoronto.ca/costanzo2009/>
- [30] A. Shojaie, G. Michailidis *Discovering graphical Granger causality using the truncating lasso penalty*, Department of Statistics, University of Michigan, ECCB 2010, Vol. 26
- [31] F. Nikolay, M. Pesavento *Learning Directed-Acyclic-Graphs from Large-Scale Double-Knockout Experiments*, C, Communications System Group, TU Darmstadt, EUSIPCO 2016, Budapest
- [32] Supplementary Material: https://www2.spg.tu-darmstadt.de/~fnikolay/supp_journal.pdf